AgentDrug: Utilizing Large Language Models in An Agentic Workflow for Zero-Shot Molecular Optimization

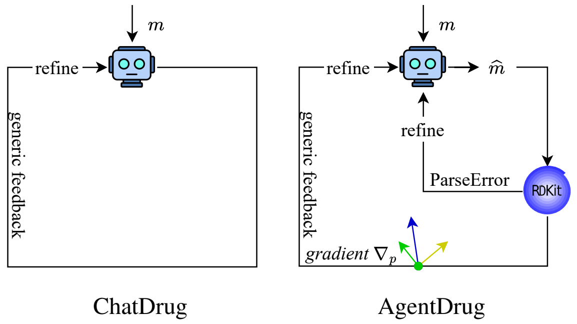
Khiem Le, Ting Hua, Nitesh V. Chawla
University of Notre Dame, IN, USA
`{kle3, thua, nchawla}@nd.edu`

Abstract

Molecular optimization—modifying a given molecule to improve desired properties—is a fundamental task in drug discovery. While LLMs hold the potential to solve this task using natural language to drive the optimization, straightforward prompting achieves limited accuracy. In this work, we propose AgentDrug [1], an agentic workflow that leverages LLMs in a structured refinement process to achieve significantly higher accuracy. AgentDrug defines a nested refinement loop: the inner loop uses feedback from cheminformatics toolkits to validate molecular structures, while the outer loop guides the LLM with generic feedback and a gradient-based objective to steer the molecule toward property improvement. We evaluate AgentDrug on benchmarks with both single- and multi-property optimization under loose and strict thresholds. Results demonstrate significant performance gains over previous methods. With Qwen-2.5-3B, AgentDrug improves accuracy by 20.7% (loose) and 16.8% (strict) on six single-property tasks, and by 7.0% and 5.3% on eight multi-property tasks. With larger model Qwen-2.5-7B, AgentDrug further improves accuracy on 6 single-property objectives by 28.9% (loose) and 29.0% (strict), and on 8 multi-property objectives by 14.9% (loose) and 13.2% (strict).

1 Introduction

The process of drug discovery, which involves identifying molecules that can safely treat or influence a disease, is expensive and typically takes over a decade to result in an approved drug (Bateman, 2022; Sertkaya et al., 2024). Recent advancements in AI for molecular studies have been revolutionizing this process by scaling many key tasks efficiently (Mak and Pichika, 2019). While significant progress has been made in molecular property prediction, synthesis, and retrosynthesis (Zhao et al., 2023; Liu et al., 2024b), molecular optimization—a crucial step in refining candidate molecules—remains relatively underexplored.



ChatDrug AgentDrug

Figure 1: An illustration of AgentDrug workflow

Molecular optimization involves modifying a given molecule to improve specific desired properties. A key challenge in this task is the similarity constraint—the modified molecule must remain structurally similar to the original (Jorgensen, 2009), which differentiates it from de novo molecular generation (Schneider and Fechner, 2005). In practice, this task is routinely performed by chemists through manual iterations (Hoffer et al., 2018), making it labor-intensive and difficult to scale. Recent breakthroughs in large language models (LLMs) offer a promising alternative: LLMs can guide molecular optimization via natural language, enabling more scalable and automated solutions. Unlike traditional supervised learning approaches (He et al., 2021, 2022), LLMs can reduce the dependence on labeled molecule pairs (Liu et al., 2023a), and exhibit zero-shot and open-vocabulary generalization beyond pre-defined objective sets. Moreover, they can also substantially reduce training costs.

Early attempts explore the use of LLMs for molecular optimization through straightforward prompting, but report limited accuracy (Zhang et al., 2024). Building on the refinement strate-

[1] https://github.com/lhkhiem28/AgentDrug

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 24448–24458
November 4-9, 2025 ©2025 Association for Computational Linguistics

gies commonly employed by chemists, Liu et al. (2024a) propose ChatDrug, a conversational framework that guides LLMs in an iterative refinement loop. This loop relies on generic feedback—for example, "The modified molecule does not meet the objective"—to prompt further revisions. However, this setup has key limitations: the generic nature of the feedback causes LLMs to refine molecules merely toward the objective, and the method does not address molecular hallucination (Guo et al., 2023), where LLMs produce invalid molecular structures. Consequently, the retrieval step is often disabled due to the high frequency of invalid outputs, diminishing its effectiveness.

In this work, we propose AgentDrug (Figure 1), an agentic workflow that addresses the limitations of previous work. AgentDrug designed a nested refinement loop to systematically improve both the validity and quality of the modified molecule through iterative interactions between the LLM and external tools. The inner loop interacts with a cheminformatics toolkit to detect and extract ParseError messages, which serve as feedback to help the LLM revise the modified molecule into a valid one. Once validity is achieved, i.e., the molecule passes the inner loop, the outer loop provides both generic feedback and an explicit gradient signal to guide the LLM in optimizing the molecule toward the target objective. In addition, AgentDrug retrieves molecules from a prepared database based on two criteria: (1) similarity to the modified molecule and (2) satisfaction of the optimization objective. These molecules serve as in-context examples to guide the LLM. With the explicit *gradient*, the LLM gains actionable guidance analogous to *gradient ascent*, enabling more effective refinement. In contrast to these earlier methods, AgentDrug enforces molecular validity early in its workflow, allowing the retrieval step to function more reliably and play a more impactful role in guiding optimization.

Table 1: Definition of the six categories of ParseError.

ParseError	Definition
syntax	The SMILES string does not follow the correct SMILES grammar, often due to unrecognized characters or patterns.
parentheses	The SMILES string contains unmatched parentheses, disrupting branching logic.
duplicate bond	The SMILES string contains a bond that is defined more than once between the same pair of atoms.
valence	The SMILES string contains an atom that is assigned more bonds than its allowed valence.
aromaticity	The SMILES string contains misused aromatic atoms, e.g., marking a non-ring atom as aromatic or causing kekulization conflicts.
unclosed ring	The SMILES string contains a ring closure digit that appears only once, meaning the ring was opened but not closed.

2 Methodology

Formally, molecular optimization is the task of modifying a given molecule m, expressed in the SMILES string (Weininger, 1988). The goal is to adjust a set of desired molecular properties $p = \{p_i\}_{i=1}^{N}$ by thresholds $d = \{d_i\}_{i=1}^{N}$. Each threshold $d_i \in \mathbb{R}$ is associated with a direction $\sigma(d_i)$, either + (indicating an increase) or - (indicating a decrease), which specifies whether the property p_i should be improved or reduced. Given an LLM $\mathcal{M}$, the optimization is defined as:

$$\widehat{m} = \mathcal{M}(p||d||m) \quad \text{s.t.} \quad \prod_{i=1}^{N} \underset{p_i,d_i}{E}(\widehat{m}) = 1 \quad \text{where}$$

$$\underset{p_i,d_i}{E}(\widehat{m}) = \mathbb{1}[\sigma(d_i)((p_i[\widehat{m}] - p_i[m]) - d_i) \geq 0].$$

Here, $||$ denotes concatenation, and $\mathbb{1} \in 0, 1$ is an indicator function that evaluates whether the change in property p_i satisfies the specified threshold d_i in the intended direction.

AgentDrug begins by utilizing LLM $\mathcal{M}$ to generate an initial modified molecule $\widehat{m}$. However, LLMs often suffer from the "molecule hallucination" phenomenon, where the generated molecule is chemically invalid. To address this, AgentDrug incorporates a validation and refinement loop inspired by debugging practices.

Specifically, the validity of $\widehat{m}$ is checked using RDKit (Landrum et al., 2013), a cheminformatics toolkit that parses the SMILES string and returns a ParseError if the molecule is invalid. There are six categories of ParseError, as summarized in Table 1. When a ParseError is detected, it is provided as feedback to the LLM, prompting it to iteratively refine $\widehat{m}$ until a valid molecule is produced.

Once the modified molecule $\widehat{m}$ is valid (it passes the inner loop), AgentDrug generates generic feedback followed by an explicit *gradient* signal ∇_p to guide further optimization:

$$\nabla_p = \{\sigma(d_i)|(p_i[\widehat{m}] - p_i[m]) - d_i|\}_{i=1}^{N}, \quad (1)$$

where $\sigma(d_i)$ denotes the desired direction of change (increase or decrease) for property p_i. The

Table 2: The results (T = 3) on a set of single- and multi-property objectives with loose and strict thresholds.

p	d	Qwen2.5-3B										Qwen2.5-7B									
		vanilla		REINVENT		ChatDrug		AgentDrug		AgentDrug†		vanilla		REINVENT		ChatDrug		AgentDrug		AgentDrug†	
		valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy
+LogP	l	68.7	21.3	66.0	17.5	65.8	27.6	**74.1**	**43.0**	68.0	32.7	76.3	29.8	75.8	29.5	72.5	40.2	**87.0**	**71.7**	73.3	46.2
	s	66.0	11.2	69.4	11.5	63.5	22.5	**72.5**	**37.7**	62.9	22.0	74.3	16.7	74.1	16.7	70.4	33.1	**84.4**	**66.2**	73.5	45.6
-LogP	l	69.0	12.4	71.4	16.1	69.0	26.6	**79.0**	**49.8**	66.4	28.2	77.8	28.8	77.2	27.8	73.5	41.2	**85.8**	**70.8**	74.9	49.1
	s	70.9	09.9	73.3	05.5	67.1	20.5	**75.8**	**40.5**	61.9	16.1	74.6	11.6	74.6	11.6	73.5	39.3	**83.0**	**63.5**	77.8	55.6
+TPSA	l	68.5	13.7	70.2	15.4	63.5	21.0	**76.6**	**46.4**	65.4	26.1	72.5	23.2	72.5	23.2	72.5	37.7	**79.4**	**57.9**	70.4	40.8
	s	67.3	08.8	67.1	09.4	63.7	20.7	**72.5**	**36.6**	61.7	18.8	69.0	20.3	69.0	20.3	66.0	27.4	**76.0**	**51.0**	67.8	35.6
-TPSA	l	68.5	10.6	69.2	11.1	66.9	23.7	**74.3**	**39.0**	64.1	21.8	73.8	11.8	73.5	11.4	73.5	36.8	**86.2**	**69.8**	77.2	52.5
	s	69.2	06.2	71.7	07.2	65.8	22.7	**76.9**	**38.8**	65.4	23.9	72.5	04.4	71.9	04.3	70.7	31.5	**82.3**	**63.4**	73.8	45.8
+QED	l	68.7	11.3	70.2	13.3	70.2	29.1	**78.7**	**48.8**	68.5	31.5	76.6	18.8	76.3	18.3	71.7	33.0	**83.7**	**65.7**	77.5	54.7
	s	70.7	02.5	70.2	04.2	66.2	10.9	**73.5**	**23.2**	64.7	12.3	79.0	05.1	78.4	04.3	74.1	20.4	**80.0**	**48.0**	78.1	41.0
-QED	l	73.5	23.9	72.2	22.0	68.0	27.2	**80.0**	**52.4**	68.0	34.4	76.9	28.5	76.6	28.4	76.9	49.6	**88.5**	**76.5**	78.7	57.5
	s	71.2	08.2	69.9	05.9	66.7	16.0	**75.2**	**37.6**	66.4	26.9	77.5	13.2	77.2	13.1	71.4	33.9	**84.4**	**67.5**	74.3	47.2
Average	l	69.5	15.5	69.9	15.9	67.2	25.9	**77.1**	**46.6**	66.7	29.1	75.6	23.5	75.3	23.1	73.4	39.8	**85.1**	**68.7**	75.3	50.1
	s	69.2	07.8	70.3	07.3	65.5	18.9	**74.4**	**35.7**	63.8	20.0	74.5	11.9	74.2	11.7	71.0	30.9	**81.7**	**59.9**	74.2	45.1
+LogP +TPSA	l l	67.1	03.4	66.4	06.3	66.2	16.9	**69.0**	**25.5**	63.9	18.2	72.7	05.1	72.2	05.0	69.7	26.8	**78.1**	**49.2**	72.7	43.6
	s s	68.0	03.4	68.0	04.4	65.6	18.7	**69.4**	**24.3**	63.3	17.7	71.2	04.6	70.9	04.6	66.7	22.0	**81.3**	**53.3**	69.9	37.8
+LogP -TPSA	l l	69.0	04.8	67.6	05.7	64.9	10.1	**69.9**	**14.7**	64.5	13.6	79.0	09.5	78.4	09.4	74.3	18.6	**81.3**	**27.2**	76.9	22.3
	s s	69.7	03.1	69.9	03.5	62.7	05.3	**70.9**	**08.2**	62.5	06.6	72.7	01.8	71.9	01.8	73.0	10.6	**80.3**	**16.1**	73.8	12.9
-LogP +TPSA	l l	69.0	04.8	67.8	07.5	68.0	16.7	**69.9**	**25.9**	65.6	19.3	78.7	26.0	78.7	26.0	72.7	39.6	**80.0**	**56.0**	76.9	52.3
	s s	67.3	02.7	70.7	03.5	63.9	10.5	**71.9**	**23.7**	66.7	18.0	76.6	13.4	76.6	13.4	67.8	25.8	**77.5**	**50.4**	72.5	42.0
-LogP -TPSA	l l	66.9	03.3	**70.4**	03.2	66.7	04.3	70.4	**10.2**	63.7	04.8	75.8	03.0	75.8	03.0	72.7	05.1	**84.0**	**14.7**	77.5	12.0
	s s	67.8	00.3	70.4	00.0	64.5	01.3	**74.9**	**02.3**	66.7	01.3	76.6	00.4	76.3	00.4	77.2	02.3	**83.7**	**07.1**	77.2	04.3
+LogP +QED	l l	66.2	07.6	69.4	09.0	65.1	12.1	**71.4**	**25.0**	65.1	18.6	74.9	06.4	74.3	06.3	73.8	31.0	**79.7**	**55.8**	76.3	50.8
	s s	66.9	01.0	**68.0**	01.0	65.6	06.2	68.0	**16.3**	65.8	09.2	72.5	01.1	72.2	01.1	71.9	14.0	**78.1**	**38.3**	74.1	33.0
+LogP -QED	l l	71.2	15.3	68.0	11.9	66.2	16.6	**71.9**	**20.5**	66.9	16.7	75.5	13.2	75.2	13.2	73.8	20.3	**84.7**	**25.4**	75.2	16.5
	s s	67.6	04.1	**69.4**	04.5	63.1	04.7	68.0	**09.5**	62.9	06.6	74.1	**08.2**	73.8	08.1	71.4	03.9	**79.0**	06.3	74.6	05.2
-LogP +QED	l l	67.6	03.4	70.4	04.9	66.7	13.3	**71.4**	**23.9**	66.4	17.6	76.9	10.0	76.3	09.2	74.1	33.0	**84.0**	**61.3**	76.6	51.3
	s s	69.2	01.4	**71.4**	01.1	65.4	07.2	71.2	**10.7**	67.6	06.8	78.4	00.4	78.1	00.0	74.6	17.9	77.8	31.9	**80.0**	**36.4**
-LogP -QED	l l	**71.9**	08.6	71.4	09.3	67.1	09.7	71.4	**10.4**	67.1	09.1	79.4	19.1	78.7	18.9	75.8	19.7	**85.1**	**24.3**	79.4	19.1
	s s	**71.7**	**03.6**	71.4	01.1	65.8	01.0	69.0	02.4	66.4	03.3	77.2	03.5	76.9	03.5	74.3	**05.6**	**81.0**	04.9	76.9	05.0
Average	l l	68.6	06.4	68.9	07.2	66.4	12.5	**70.7**	**19.5**	65.4	14.7	76.6	11.5	76.2	11.4	73.4	24.3	**82.1**	**39.2**	76.4	33.5
	s s	68.5	02.5	69.9	02.4	64.6	06.9	**70.4**	**12.2**	65.2	08.7	74.9	04.2	74.6	04.1	72.1	12.8	**79.8**	**26.0**	74.9	22.1

p	d	Llama-3.1-8B										Llama-3.1-70B									
		vanilla		REINVENT		ChatDrug		AgentDrug		AgentDrug†		vanilla		REINVENT		ChatDrug		AgentDrug		AgentDrug†	
		valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy	valid	accuracy
+LogP	l	76.3	40.1	73.3	32.9	76.0	48.3	**85.8**	**69.1**	76.6	52.9	82.0	57.4	81.4	56.9	83.7	67.4	**94.8**	**89.1**	82.6	65.3
	s	71.7	26.2	75.4	26.8	71.4	38.6	**81.6**	**61.6**	71.9	41.4	87.7	35.1	87.4	35.0	80.3	59.8	**92.2**	**81.6**	83.3	65.8
-LogP	l	82.0	25.8	**84.8**	33.5	75.8	36.4	83.0	**56.8**	74.9	43.4	82.6	55.0	82.0	53.0	78.4	56.1	**91.7**	**83.5**	79.0	57.7
	s	79.0	13.4	81.7	07.5	72.7	27.3	**87.0**	**60.9**	72.2	36.8	82.0	30.3	82.0	30.2	77.8	52.9	**94.8**	**84.8**	79.0	56.5
+TPSA	l	75.2	25.9	77.0	29.2	74.6	42.5	**85.8**	**70.4**	72.5	43.8	80.6	59.7	80.6	59.6	80.0	59.6	**90.1**	**80.2**	80.0	60.0
	s	74.9	24.0	74.7	25.6	74.6	44.0	**85.8**	**70.0**	73.8	46.9	79.0	55.3	79.0	55.4	79.4	58.7	**93.9**	**86.8**	82.0	63.9
-TPSA	l	78.4	16.5	79.2	17.2	74.9	41.6	**89.7**	**75.8**	76.3	49.6	85.1	37.0	84.8	35.8	80.0	59.2	**96.6**	**90.8**	84.7	68.6
	s	82.0	17.2	84.9	19.9	77.2	40.9	**86.6**	**66.7**	74.1	45.9	86.6	31.6	85.9	31.0	77.5	53.1	**95.2**	**89.5**	83.7	66.1
+QED	l	80.0	18.4	81.7	21.7	75.2	39.9	**85.1**	**60.4**	74.9	43.8	86.6	34.6	86.3	33.7	81.3	52.8	**93.5**	**82.7**	83.3	62.5
	s	80.6	02.8	80.0	04.7	74.6	17.9	**83.0**	**45.6**	75.8	28.0	90.1	06.3	89.4	05.3	85.1	36.6	**94.3**	**62.7**	84.7	45.8
-QED	l	77.5	31.8	76.1	29.3	73.0	39.4	**86.6**	**68.8**	75.5	49.8	80.6	41.5	80.3	41.3	77.8	55.3	**94.3**	**87.3**	79.4	58.7
	s	81.6	22.4	80.2	16.3	71.9	34.2	**85.5**	**66.2**	69.7	34.8	87.0	06.1	86.6	06.1	78.7	52.4	**92.6**	**80.1**	81.0	58.3
Average	l	78.2	26.4	78.7	27.0	74.9	41.3	**86.0**	**66.9**	75.1	47.2	82.9	47.5	82.6	46.8	80.2	58.4	**93.5**	**85.6**	81.5	62.1
	s	78.3	17.7	79.5	16.5	73.8	33.8	**84.9**	**61.8**	72.9	39.0	85.4	27.5	85.1	27.1	79.8	52.3	**93.8**	**80.9**	82.3	59.4
+LogP +TPSA	l l	73.3	11.0	72.5	20.4	67.1	23.8	**73.8**	**42.4**	69.7	33.5	84.7	17.4	84.1	17.2	74.3	36.4	**88.1**	**55.9**	81.0	45.3
	s s	69.2	10.0	69.2	13.0	66.4	19.6	**71.7**	31.9	70.2	**32.3**	84.4	10.6	84.0	10.6	76.3	34.7	**87.0**	**57.0**	82.0	48.8
+LogP -TPSA	l l	76.3	11.8	74.7	14.1	71.2	19.6	**78.4**	**27.1**	72.5	24.6	85.8	41.6	85.2	41.2	83.7	48.1	**91.3**	**58.9**	83.3	47.5
	s s	74.6	06.0	74.9	06.7	70.4	08.8	**80.6**	**16.1**	70.7	11.3	**91.3**	21.5	90.4	21.5	79.7	30.3	**91.3**	**33.3**	83.7	28.4
-LogP +TPSA	l l	73.8	11.4	72.5	17.8	71.2	19.2	**78.4**	**42.0**	70.2	31.9	83.0	57.7	83.0	57.6	79.4	57.1	**91.3**	**80.4**	84.0	65.5
	s s	70.4	06.7	73.9	08.7	70.9	14.5	**75.5**	29.8	70.7	**30.4**	87.0	37.0	87.0	37.0	78.7	53.5	**92.6**	**74.1**	84.7	57.2
-LogP -TPSA	l l	77.5	03.1	**81.6**	03.0	71.9	**09.7**	79.7	09.6	74.9	08.2	81.0	02.8	80.9	02.9	70.2	05.3	**85.8**	**09.4**	74.9	06.7
	s s	76.3	00.8	79.3	00.0	73.0	05.1	**80.6**	**06.0**	75.2	05.3	81.3	02.0	81.0	01.9	75.2	**06.4**	**88.9**	06.2	77.8	05.1
+LogP +QED	l l	81.0	13.4	**84.9**	15.9	73.0	20.4	83.0	**46.0**	71.9	29.1	84.4	20.7	83.7	20.4	78.4	35.7	**87.7**	**47.8**	78.7	40.9
	s s	80.0	03.2	**81.3**	03.3	71.2	07.5	74.6	**23.9**	74.1	18.5	**89.3**	02.2	88.9	02.2	82.0	20.1	88.5	**36.3**	85.5	26.9
+LogP -QED	l l	76.0	25.9	72.6	20.1	74.9	28.5	**77.2**	**35.5**	71.4	24.3	84.7	53.0	84.4	52.8	82.3	53.5	**91.3**	**65.3**	84.7	55.5
	s s	86.2	15.5	**88.5**	17.1	81.0	**22.7**	75.8	16.3	76.6	15.3	89.7	12.6	89.3	12.4	85.1	**21.3**	**90.9**	12.7	88.9	14.2
-LogP +QED	l l	78.4	08.2	**81.7**	11.9	77.2	21.6	77.5	**32.9**	75.2	29.3	80.6	16.9	80.0	15.5	74.6	23.9	**87.0**	**38.7**	79.0	32.0
	s s	77.8	01.2	**80.3**	00.9	76.0	07.2	78.1	**21.5**	77.8	18.7	85.5	02.6	85.2	00.0	81.0	13.8	**85.8**	**28.8**	84.4	21.9
-LogP -QED	l l	77.8	13.2	77.3	14.3	75.5	14.3	**81.3**	**14.6**	73.5	11.0	78.4	34.1	77.8	33.8	75.5	35.9	**84.4**	**48.9**	74.9	33.0
	s s	83.0	24.5	82.7	07.3	78.1	22.6	**86.2**	**26.3**	81.6	22.4	84.0	07.1	83.7	07.1	73.3	08.8	**88.9**	**11.1**	80.6	08.1
Average	l l	76.8	12.3	77.1	13.8	72.7	19.6	**78.7**	**31.3**	72.4	24.0	82.8	30.5	82.4	30.1	77.3	37.0	**88.4**	**50.7**	80.1	40.8
	s s	77.2	08.5	**78.8**	08.3	73.4	13.5	77.9	**21.5**	74.6	19.3	86.5	11.9	86.2	11.7	78.9	23.6	**89.2**	**32.4**	83.4	26.3

gradient encodes both the direction and the magnitude of required adjustments to the desired properties p, and is used to explicitly steer the LLM $\mathcal{M}$ in refining $\widehat{m}$ toward the optimization objective.

Additionally, AgentDrug retrieves a molecule m_e from a prepared database $\mathcal{D}$. This molecule is selected for its similarity to the modified molecule $\widehat{m}$ and its ability to meet the optimization objective:

$$m_e = \underset{m'_e \in \mathcal{D}}{\operatorname{argmax}} \left\langle m'_e, \widehat{m} \right\rangle \wedge \prod_{i=1}^{N} \underset{p_i, d_i}{E} \left(m'_e \right) \quad (2)$$

where $\langle m'_e, \widehat{m} \rangle$ represents the Tanimoto similarity (Bajusz et al., 2015) between m'_e and $\widehat{m}$. The molecule m_e serves as an in-context example, helping the LLM $\mathcal{M}$ refine $\widehat{m}$ toward the desired properties. By ensuring $\widehat{m}$ is valid, the retrieval step becomes more efficient and effective.

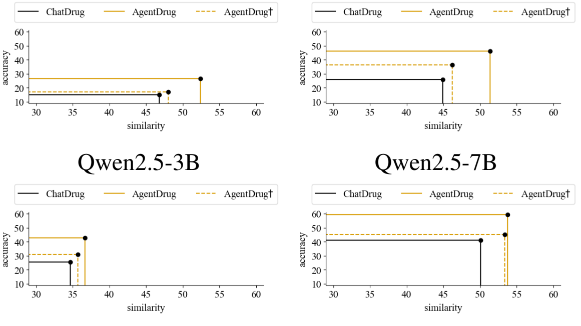
3 Evaluation

Following previous work (Liu et al., 2024a), we select 500 molecules from the ZINC database (Irwin et al., 2012) as input molecules. Three molecular properties—LogP, TPSA, and QED—are used as optimization targets. We evaluate performance

Table 3: The results (T = 3) on a set of single- and multi-property objectives with loose and strict thresholds.

p	d	Qwen2.5-3B			Qwen2.5-7B		
		ChatDrug	AgentDrug	AgentDrug†	ChatDrug	AgentDrug	AgentDrug†
		similarity	similarity	similarity	similarity	similarity	similarity
+LogP	l	53.7	54.7	**55.2**	56.2	**60.2**	57.1
	s	51.6	49.5	**53.7**	46.1	**53.1**	47.3
−LogP	l	53.6	**57.6**	55.2	53.9	**59.6**	58.0
	s	48.2	**53.5**	47.3	46.5	**50.5**	44.9
+TPSA	l	48.8	**56.5**	54.0	55.1	**56.8**	52.1
	s	49.5	52.2	**53.0**	51.2	**56.7**	50.9
−TPSA	l	44.7	**50.9**	48.3	45.1	**49.7**	45.6
	s	42.8	**46.5**	40.7	39.6	**45.2**	40.2
+QED	l	48.8	**56.6**	51.1	47.6	**56.6**	51.0
	s	46.3	**55.3**	50.6	42.1	**56.6**	44.9
−QED	l	54.2	**63.0**	55.5	55.1	**56.4**	56.2
	s	45.6	**56.3**	47.5	42.5	**48.5**	40.1
Average	l	50.6	**56.6**	53.2	52.2	**56.6**	53.3
	s	47.3	**52.2**	48.8	44.7	**51.8**	44.7
+LogP +TPSA	l l	43.3	**53.2**	44.9	36.7	**45.3**	41.0
	s s	37.9	**41.2**	40.3	30.6	**41.5**	36.8
+LogP −TPSA	l l	46.4	**58.3**	47.6	45.2	**50.7**	46.0
	s s	46.1	**48.8**	43.6	42.3	**48.8**	43.5
−LogP +TPSA	l l	49.7	**52.3**	46.5	51.8	**54.4**	50.6
	s s	45.5	**49.3**	43.4	38.5	**47.9**	40.1
−LogP −TPSA	l l	41.1	**51.6**	40.9	42.3	**47.5**	44.5
	s s	46.1	**47.3**	44.9	40.1	**49.8**	41.8
+LogP +QED	l l	45.9	**53.5**	47.7	41.0	**52.2**	43.7
	s s	43.3	**50.5**	48.2	43.7	**54.6**	42.6
+LogP −QED	l l	50.2	**58.7**	50.6	47.2	**51.2**	49.1
	s s	44.8	**50.5**	43.3	41.6	**47.2**	45.1
−LogP +QED	l l	42.7	**52.1**	46.3	40.1	**46.9**	42.0
	s s	46.7	50.3	**52.8**	45.8	**54.6**	45.3
−LogP −QED	l l	44.8	**52.6**	46.4	48.7	**50.9**	50.7
	s s	**47.0**	43.1	44.0	41.1	**46.5**	42.9
Average	l l	45.5	**54.0**	46.4	44.1	**49.9**	46.0
	s s	44.7	**47.6**	45.1	40.5	**48.9**	42.3

p	d	Llama-3.1-8B			Llama-3.1-70B		
		ChatDrug	AgentDrug	AgentDrug†	ChatDrug	AgentDrug	AgentDrug†
		similarity	similarity	similarity	similarity	similarity	similarity
+LogP	l	44.1	**47.3**	46.0	68.7	**70.7**	69.7
	s	37.9	39.1	**40.9**	51.4	**54.9**	53.3
−LogP	l	38.8	**46.2**	42.3	69.0	**70.9**	69.1
	s	33.2	36.2	**37.3**	51.6	**56.5**	55.8
+TPSA	l	41.8	**45.0**	44.9	70.5	70.8	**71.0**
	s	37.6	39.1	**40.6**	66.8	67.8	**68.1**
−TPSA	l	37.7	**38.0**	35.2	52.7	53.3	**53.8**
	s	**31.3**	31.0	29.3	45.2	44.4	**45.3**
+QED	l	40.6	41.0	**41.4**	46.6	51.5	**52.6**
	s	32.5	**38.5**	34.1	45.7	**46.6**	44.4
−QED	l	46.7	**47.1**	45.4	64.0	66.4	**67.1**
	s	33.4	**34.1**	34.1	40.2	44.4	**45.2**
Average	l	41.6	**44.1**	42.5	61.9	**63.9**	63.9
	s	34.3	**36.3**	36.1	50.2	**52.4**	52.0
+LogP +TPSA	l l	**34.1**	32.9	32.4	39.7	48.1	**48.1**
	s s	29.0	**32.3**	27.0	29.7	36.9	**39.5**
+LogP −TPSA	l l	**37.6**	35.5	37.0	49.5	**51.9**	51.5
	s s	31.1	32.0	**32.4**	41.9	**45.2**	44.1
−LogP +TPSA	l l	35.3	**38.0**	36.3	65.0	**69.0**	66.2
	s s	31.7	**34.0**	31.5	52.9	**55.3**	54.4
−LogP −TPSA	l l	33.3	35.9	**36.6**	41.4	**47.6**	44.7
	s s	32.8	**34.3**	30.5	36.8	40.3	**40.5**
+LogP +QED	l l	36.1	36.4	**37.0**	45.0	46.4	**48.3**
	s s	**36.1**	35.4	32.7	42.5	45.1	**48.3**
+LogP −QED	l l	31.5	36.1	**36.4**	57.1	**59.6**	58.7
	s s	20.8	24.3	**24.9**	42.3	**48.8**	48.3
−LogP +QED	l l	37.7	**40.1**	37.6	42.5	**52.9**	52.1
	s s	32.0	**36.2**	35.2	44.9	**51.4**	47.2
−LogP −QED	l l	36.0	**41.6**	38.1	58.5	62.5	**63.0**
	s s	21.4	20.8	**22.7**	41.5	**47.2**	45.3
Average	l l	35.2	**37.1**	36.4	49.8	**54.8**	54.1
	s s	29.4	**31.2**	29.6	41.6	**46.3**	45.9



Qwen2.5-3B

Qwen2.5-7B

Llama-3.1-8B

Llama-3.1-70B

Figure 2: Comparison of accuracy vs. similarity trade-offs for different models. Each plot shows the performance of ChatDrug (black), AgentDrug (yellow solid), and AgentDrug† (yellow dashed) across four LLMs.

on both single-property and multi-property optimization tasks under two levels of difficulty: loose and strict thresholds, which define the required amounts of property increase or decrease (see Table 4 for threshold values). We evaluate our results on four open-source LLMs: Qwen-2.5 (Yang et al., 2024) (3B and 7B) and LLaMA-3.1 (Dubey et al., 2024) (8B and 70B). As baselines, we include straightforward prompting (vanilla LLM response without iterative refinement) and a reinforcement learning-based method, REINVENT (Olivecrona et al., 2017). More experiment settings are shown in the Appendix.

Tables 2 and 3 report results on single- and multi-property optimization tasks under both loose (l) and strict (s) thresholds, using three refinement iterations ($T = 3$). First, thanks to its inner loop, AgentDrug significantly reduces the molecular hallucination, increasing the likelihood that the modified molecule is chemically valid. The performance gain of AgentDrug over its ablated version, AgentDrug† (without the inner loop), directly demonstrates the effectiveness of this component.

The results show that AgentDrug consistently outperforms ChatDrug. With Qwen-2.5-3B, AgentDrug improves accuracy by 20.7% (loose) and 16.8% (strict) on six single-property objectives, and by 7.0% and 5.3% on eight multi-property objectives. With the larger Qwen-2.5-7B, the gains increase to 28.9% and 29.0% (single-property), and 14.9% and 13.2% (multi-property), under loose and strict thresholds, respectively. Notably, AgentDrug† also consistently outperforms ChatDrug, highlighting the value of the explicit *gradient* signal, even without the inner loop. While straightforward prompting achieves limited accuracy, the REINVENT baseline also fails to yield meaningful improvements—despite incurring high training costs. This is largely because REINVENT requires a supervised pretraining phase with labeled molecule pairs, which are often unavailable in practice.

Beyond accuracy, AgentDrug is also more effective in preserving the similarity constraint. As illustrated in Figure 2, both AgentDrug and AgentDrug† explore molecular space more effectively than ChatDrug, achieving better optimization while maintaining similarity, which further demonstrates the effectiveness of the *gradient*-guided refinement.

4 Ablation Studies

4.1 Impact of using the example molecule

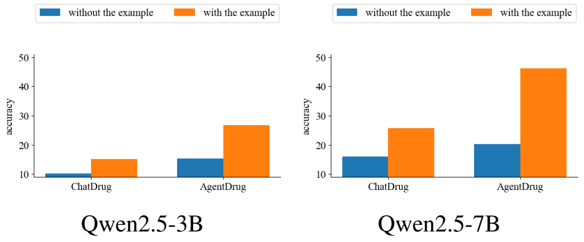
We conduct an ablation study to evaluate the impact of incorporating the retrieved example molecule m_e, a component that is overlooked in prior work (Liu et al., 2024a). Table 5 presents results for both AgentDrug and ChatDrug when m_e is excluded. Even without m_e, AgentDrug/m_e consistently outperforms ChatDrug/m_e, demonstrating the effectiveness of the remaining components in AgentDrug. Using Qwen-2.5-3B, AgentDrug/m_e improves accuracy by 11.2% and 6.7% under loose and strict thresholds, respectively, on six single-property objectives, and by 3.0% and 1.4% on eight multi-property objectives. With Qwen-2.5-7B, improvements are 10.1% and 5.8% (single-property), and 2.4% and 0.9% (multi-property), under loose and strict thresholds, respectively.

Moreover, Figure 3 visually illustrates the contribution of incorporating the example molecule m_e in both AgentDrug and ChatDrug. As discussed earlier, AgentDrug significantly reduces the molecular hallucination problem, increasing the likelihood that the modified molecule is valid. This, in turn, enables more reliable use of the retrieval step. By ensuring molecular validity, AgentDrug is able to fully leverage the retrieved example molecule m_e, thereby maximizing its effectiveness in guiding the optimization process.

4.2 Impact of the number of iterations

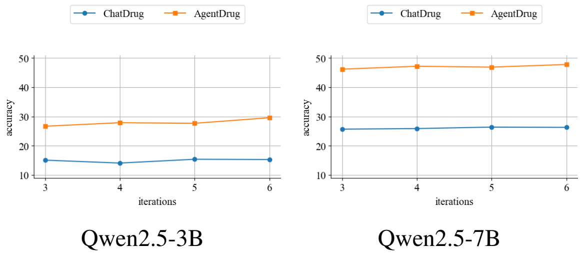
We conduct an ablation study to examine how AgentDrug and ChatDrug behave under different numbers of refinement iterations T. Table 6 presents results for both methods using $T = 4$, 5, and 6 iterations, respectively. The results show that AgentDrug consistently outperforms ChatDrug when using the same number of iterations. Notably, AgentDrug with fewer iterations still surpasses ChatDrug with more iterations, highlighting the superior efficiency and effectiveness of the AgentDrug framework.

However, as shown in Figure 4, increasing the number of iterations does not lead to substantial performance gains for either AgentDrug or ChatDrug, indicating a saturation point in their effectiveness.



Qwen2.5-3B Qwen2.5-7B

Figure 3: Impact of using the example molecule. Averaged results on all objectives.



Qwen2.5-3B Qwen2.5-7B

Figure 4: Impact of the number of iterations. Averaged results on all objectives.

5 Conclusion

This paper introduced AgentDrug, a novel approach leveraging LLMs for zero-shot molecular optimization through gradient-guided refinement. By providing explicit gradient information during the refinement process, AgentDrug enables targeted molecular modifications while maintaining similarity constraints. Our comprehensive experiments show that AgentDrug consistently outperforms powerful baselines on both single- and multi-property optimization tasks. It is particularly effective for complex multi-property objectives and can balance well with optimization accuracy and molecular similarity.

Limitations

First of all, although it does not require any training, we admit that utilizing LLMs in a loop of refinement requires prompting LLMs over multiple iterations, obviously leading to multiplying prompting costs (Samsi et al., 2023). However, this limitation is acceptable and compensated for by bringing improved performance. In addition, albeit containing a *gradient* with directions and magnitudes to explicitly guide LLMs to refine the modified molecule toward the objective, feedback from AgentDrug currently lacks concrete actions towards the objective, which involve knowledge of molecular properties (Fang et al., 2023; Hoang et al., 2024), are presumably useful but hard to establish.

References

Dávid Bajusz, Anita Rácz, and Károly Héberger. 2015. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of cheminformatics*, 7:1–13.

Thomas J. Bateman. 2022. Chapter 29 - drug discovery. In Shiew-Mei Huang, Juan J.L. Lertora, Paolo Vicini, and Arthur J. Atkinson, editors, *Atkinson's Principles of Clinical Pharmacology (Fourth Edition)*, fourth edition edition, pages 563–572. Academic Press, Boston.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. 2022. Translation between molecules and natural language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 375–413, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yin Fang, Qiang Zhang, Ningyu Zhang, Zhuo Chen, Xiang Zhuang, Xin Shao, Xiaohui Fan, and Huajun Chen. 2023. Knowledge graph-enhanced molecular contrastive learning with functional prompt. *Nature Machine Intelligence*, 5(5):542–553.

Taicheng Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh Chawla, Olaf Wiest, Xiangliang Zhang, et al. 2023. What can large language models do in chemistry? a comprehensive benchmark on eight tasks. *Advances in Neural Information Processing Systems*, 36:59662–59688.

Jiazhen He, Eva Nittinger, Christian Tyrchan, Werngard Czechtizky, Atanas Patronov, Esben Jannik Bjerrum, and Ola Engkvist. 2022. Transformer-based molecular optimization beyond matched molecular pairs. *Journal of cheminformatics*, 14(1):18.

Jiazhen He, Huifang You, Emil Sandström, Eva Nittinger, Esben Jannik Bjerrum, Christian Tyrchan, Werngard Czechtizky, and Ola Engkvist. 2021. Molecular optimization by capturing chemist's intuition using deep neural networks. *Journal of cheminformatics*, 13:1–17.

Thanh Lam Hoang, Marco Luca Sbodio, Marcos Martinez Galindo, Mykhaylo Zayats, Raul Fernandez-Diaz, Victor Valls, Gabriele Picco, Cesar Berrospi, and Vanessa Lopez. 2024. Knowledge enhanced representation learning for drug discovery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 10544–10552.

Laurent Hoffer, Yuliia V Voitovich, Brigitt Raux, Kendall Carrasco, Christophe Muller, Aleksey Y Fedorov, Carine Derviaux, Agnès Amouric, Stéphane Betzi, Dragos Horvath, et al. 2018. Integrated strategy for lead optimization based on fragment growing: the diversity-oriented-target-focused-synthesis approach. *Journal of medicinal chemistry*, 61(13):5719–5732.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. 2012. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768.

William L Jorgensen. 2009. Efficient drug lead discovery and optimization. *Accounts of chemical research*, 42(6):724–733.

Greg Landrum et al. 2013. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling. *Greg Landrum*, 8(31.10):5281.

Khiem Le, Zhichun Guo, Kaiwen Dong, Xiaobao Huang, Bozhao Nan, Roshni Iyer, Xiangliang Zhang, Olaf Wiest, Wei Wang, and Nitesh V Chawla. 2024. Molx: Enhancing large language models for molecular learning with a multi-modal extension. *arXiv preprint arXiv:2406.06777*.

Jiatong Li, Yunqing Liu, Wenqi Fan, Xiao-Yong Wei, Hui Liu, Jiliang Tang, and Qing Li. 2024. Empowering molecule discovery for molecule-caption translation with large language models: A chatgpt perspective. *IEEE transactions on knowledge and data engineering*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Shengchao Liu, Jiongxiao Wang, Yijin Yang, Chengpeng Wang, Ling Liu, Hongyu Guo, and Chaowei Xiao. 2024a. Conversational drug editing using retrieval and domain feedback. In *The Twelfth International Conference on Learning Representations*.

Zhiyuan Liu, Sihang Li, Yanchen Luo, Hao Fei, Yixin Cao, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. 2023b. MolCA: Molecular graph-language modeling with cross-modal projector and uni-modal adapter. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15623–15638, Singapore. Association for Computational Linguistics.

Zhiyuan Liu, Yaorui Shi, An Zhang, Sihang Li, Enzhi Zhang, Xiang Wang, Kenji Kawaguchi, and Tat-Seng Chua. 2024b. ReactXT: Understanding molecular "reaction-ship" via reaction-contextualized molecule-text pretraining. In *Findings of the Association for*

Computational Linguistics ACL 2024, pages 5353–5377, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Kit-Kay Mak and Mallikarjuna Rao Pichika. 2019. Artificial intelligence in drug development: present status and future prospects. *Drug discovery today*, 24(3):773–780.

Yasmine Nahal, Janosch Menke, Julien Martinelli, Markus Heinonen, Mikhail Kabeshov, Jon Paul Janet, Eva Nittinger, Ola Engkvist, and Samuel Kaski. 2024. Human-in-the-loop active learning for goal-oriented molecule generation. *Journal of Cheminformatics*, 16(1):1–24.

Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. 2017. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics*, 9:1–14.

Siddharth Samsi, Dan Zhao, Joseph McDonald, Baolin Li, Adam Michaleas, Michael Jones, William Bergeron, Jeremy Kepner, Devesh Tiwari, and Vijay Gadepally. 2023. From words to watts: Benchmarking the energy costs of large language model inference. In *2023 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–9. IEEE.

Gisbert Schneider and Uli Fechner. 2005. Computer-based de novo design of drug-like molecules. *Nature reviews Drug discovery*, 4(8):649–663.

Philipp Seidl, Andreu Vall, Sepp Hochreiter, and Günter Klambauer. 2023. Enhancing activity prediction models in drug discovery with the ability to understand human language. In *International Conference on Machine Learning*, pages 30458–30490. PMLR.

Aylin Sertkaya, Trinidad Beleche, Amber Jessup, and Benjamin D Sommers. 2024. Costs of drug development and research and development intensity in the us, 2000-2018. *JAMA network open*, 7(6):e2415445–e2415445.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12.

Sally Turutov and Kira Radinsky. 2024. Molecular optimization model with patentability constraint. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 257–264.

David Weininger. 1988. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36.

Zhenxing Wu, Odin Zhang, Xiaorui Wang, Li Fu, Huifeng Zhao, Jike Wang, Hongyan Du, Dejun Jiang, Yafeng Deng, Dongsheng Cao, et al. 2024. Leveraging language model for advanced multiproperty molecular optimization via prompt engineering. *Nature Machine Intelligence*, pages 1–11.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Jinlu Zhang, Yin Fang, Xin Shao, Huajun Chen, Ningyu Zhang, and Xiaohui Fan. 2024. The future of molecular studies through the lens of large language models. *Journal of Chemical Information and Modeling*, 64(3):563–566.

Haiteng Zhao, Shengchao Liu, Ma Chang, Hannan Xu, Jie Fu, Zhihong Deng, Lingpeng Kong, and Qi Liu. 2023. Gimlet: A unified graph-text model for instruction-based molecule zero-shot learning. *Advances in Neural Information Processing Systems*, 36:5850–5887.

A Evaluation Settings

Each optimization objective is expressed in natural language using a standardized prompt template (see Box A1). Specifically, for each objective, we sample 1,000 molecules from the ZINC database and train LLMs using the REINFORCE algorithm (Sutton et al., 1999), where the reward function is defined as $\prod_{i=1}^{N} \underset{p_i,d_i}{E}(\widehat{m})$. The models are fine-tuned using LoRA (Hu et al., 2022) with a rank of 16 and a learning rate of 1×10^{-5}, for a single training epoch.

Table 4: The set of single- and multi-property objectives with loose and strict thresholds is used.

p	d	
+LogP	l	+00.0
	s	+00.5
-LogP	l	-00.0
	s	-00.5
+TPSA	l	+00.0
	s	+10.0
-TPSA	l	-00.0
	s	-10.0
+QED	l	+00.0
	s	+00.1
-QED	l	-00.0
	s	-00.1
Average	l	-
	s	-
+LogP +TPSA	l l	+00.0 +00.0
	s s	+00.5 +10.0
+LogP -TPSA	l l	+00.0 -00.0
	s s	+00.5 -10.0
-LogP +TPSA	l l	-00.0 +00.0
	s s	-00.5 +10.0
-LogP -TPSA	l l	-00.0 -00.0
	s s	-00.5 -10.0
+LogP +QED	l l	+00.0 +00.0
	s s	+00.5 +00.1
+LogP -QED	l l	+00.0 -00.0
	s s	+00.5 -00.1
-LogP +QED	l l	-00.0 +00.0
	s s	-00.5 +00.1
-LogP -QED	l l	-00.0 -00.0
	s s	-00.5 -00.1
Average	l l	-
	s s	-

Box A1: Template for wrapping an objective in natural language to form the prompt

Given [a given molecule m], modify it to [increase or decrease] its [desired properties $\{p_i\}_{i=1}^{N}$] by [thresholds $\{|d_i|\}_{i=1}^{N}$], respectively. Importantly, the modified molecule must be similar to the given one.

Respond with only the SMILES string of the modified molecule. No explanation is needed.

Table 5: The results (T = 3) on a set of single- and multi-property objectives with loose and strict thresholds.

p	d	Qwen2.5-3B		Qwen2.5-7B	
		ChatDrug accuracy	AgentDrug accuracy	ChatDrug accuracy	AgentDrug accuracy
+LogP	l	25.3	**39.3**	26.1	**49.0**
	s	18.5	**28.8**	18.1	**28.6**
-LogP	l	21.6	**27.7**	39.7	**45.9**
	s	09.7	**15.7**	24.8	**25.8**
+TPSA	l	16.8	**28.7**	24.7	**36.5**
	s	15.8	**29.5**	22.7	**35.1**
-TPSA	l	14.5	**22.0**	**20.4**	18.1
	s	**13.9**	13.1	**12.3**	10.8
+QED	l	19.6	**25.2**	23.7	**29.5**
	s	05.6	**06.3**	**05.9**	05.2
-QED	l	21.6	**44.0**	38.4	**54.2**
	s	07.9	**18.4**	16.3	**29.3**
Average	l	19.9	**31.1**	28.8	**38.9**
	s	11.9	**18.6**	16.7	**22.5**
+LogP +TPSA	l l	06.5	**10.2**	08.0	**11.2**
	s s	01.6	**07.4**	05.6	**09.6**
+LogP -TPSA	l l	07.2	**11.6**	**11.8**	09.6
	s s	**06.2**	05.3	**08.1**	03.6
-LogP +TPSA	l l	09.3	**15.8**	24.6	**35.2**
	s s	03.8	**08.4**	13.3	**21.8**
-LogP -TPSA	l l	**06.3**	05.0	**08.0**	02.9
	s s	**01.3**	00.7	**02.8**	01.1
+LogP +QED	l l	09.6	**12.3**	12.2	**13.6**
	s s	**02.2**	01.6	01.3	**01.8**
+LogP -QED	l l	13.3	**23.3**	21.9	**27.7**
	s s	05.8	**07.6**	**08.0**	07.9
-LogP +QED	l l	08.3	**09.1**	**14.2**	13.5
	s s	**01.3**	00.7	01.4	**01.5**
-LogP -QED	l l	**11.2**	09.0	25.0	**31.2**
	s s	03.0	**04.4**	07.8	**08.7**
Average	l l	09.0	**12.0**	15.7	**18.1**
	s s	03.1	**04.5**	06.1	**07.0**

B Ablation Studies

B.1 Impact of using the example molecule

Table 5 presents results for both AgentDrug and ChatDrug when m_e is excluded.

B.2 Impact of the number of iterations

Table 6 presents results for both methods using $T = 4$, 5, and 6 iterations, respectively.

C Ablation for the Database

In our main experiments, the database $\mathcal{D}$ consists of 10K molecules. Here, we conducted additional experiments with smaller (5K) and larger (20K) databases to thoroughly observe the behavior of AgentDrug. Table 7 displays the results of AgentDrug (with various sizes of the database $\mathcal{D}$) on LogP and TPSA using Qwen2.5 3B and 7B with loose thresholds.

We can observe that the size of the database $\mathcal{D}$ has a nonlinear impact on the performance of AgentDrug, and the impact depends on target properties. This might indicate that the property distributions of molecules in the database are more crucial than the size. Importantly, AgentDrug still performs well with a small database (5K), demonstrating its robustness.

Table 6: The results (T = 4, 5, 6) on a set of single- and multi-property objectives with loose and strict thresholds.

p	*d*	Qwen2.5-3B						Qwen2.5-7B					
		ChatDrug-4	AgentDrug-4	ChatDrug-5	AgentDrug-5	ChatDrug-6	AgentDrug-6	ChatDrug-4	AgentDrug-4	ChatDrug-5	AgentDrug-5	ChatDrug-6	AgentDrug-6
		accuracy	accuracy	accuracy	accuracy	accuracy	accuracy	accuracy	accuracy	accuracy	accuracy	accuracy	accuracy
+LogP	l	28.2	**53.5**	28.1	**49.6**	31.5	**54.5**	45.2	**68.8**	43.8	**71.5**	41.6	**73.0**
	s	20.4	**41.5**	25.8	**41.5**	22.4	**47.5**	32.1	**65.0**	34.3	**65.0**	32.1	**72.2**
-LogP	l	19.9	**50.6**	26.1	**45.6**	25.5	**53.8**	40.9	**70.4**	41.9	**70.8**	38.2	**76.4**
	s	18.3	**38.0**	20.6	**38.1**	19.9	**47.0**	39.6	**71.4**	38.1	**73.8**	39.9	**71.4**
+TPSA	l	21.6	**47.7**	23.9	**48.3**	21.4	**49.6**	36.1	**70.5**	37.1	**58.6**	40.0	**62.9**
	s	21.9	**41.5**	23.6	**44.1**	26.1	**48.5**	33.4	**53.1**	27.9	**52.9**	29.6	**55.3**
-TPSA	l	21.8	**41.5**	24.2	**41.1**	25.6	**45.0**	36.0	**66.8**	33.0	**72.8**	39.5	**72.6**
	s	24.3	**41.8**	21.5	**40.3**	27.4	**45.5**	30.8	**58.4**	36.9	**71.0**	38.9	**66.4**
+QED	l	22.6	**44.8**	24.7	**48.6**	23.8	**55.1**	34.2	**71.8**	40.1	**69.6**	39.0	**77.2**
	s	08.0	**20.2**	13.3	**23.2**	11.3	**25.0**	22.2	**53.5**	26.6	**51.0**	20.7	**47.8**
-QED	l	28.5	**61.9**	34.0	**63.4**	29.6	**54.8**	44.6	**72.4**	45.9	**81.3**	46.3	**74.2**
	s	13.1	**44.0**	17.2	**44.2**	15.8	**50.0**	33.8	**64.7**	38.2	**64.9**	33.8	**64.9**
Average	l	23.8	**50.0**	26.8	**49.4**	26.2	**52.1**	39.5	**70.1**	40.3	**70.8**	40.8	**72.7**
	s	17.7	**37.8**	20.3	**38.6**	20.5	**43.9**	32.0	**61.0**	33.7	**63.1**	32.5	**63.0**
+LogP +TPSA	l l	15.4	**24.7**	15.4	**25.0**	15.4	**27.1**	26.8	**60.8**	26.8	**51.4**	26.8	**52.8**
	s s	16.9	**22.7**	16.9	**23.3**	16.9	**25.5**	22.0	**50.2**	22.0	**46.1**	22.0	**49.4**
+LogP -TPSA	l l	08.8	**15.7**	08.8	**12.8**	08.8	**14.0**	18.6	**27.7**	18.6	**27.1**	18.6	**30.6**
	s s	03.5	**07.9**	03.5	**06.9**	03.5	**06.9**	10.6	**16.5**	10.6	**14.1**	10.6	**18.1**
-LogP +TPSA	l l	15.7	**22.6**	15.7	**24.5**	15.7	**22.6**	39.6	**53.8**	39.6	**56.0**	39.6	**58.0**
	s s	10.1	**22.8**	10.1	**24.7**	10.1	**24.7**	25.8	**50.2**	25.8	**54.2**	25.8	**54.2**
-LogP -TPSA	l l	05.1	**06.2**	05.1	**08.2**	05.1	**08.8**	05.1	**12.6**	05.1	**12.8**	05.1	**12.4**
	s s	01.3	**01.4**	01.3	**02.8**	01.3	**01.7**	02.3	**05.8**	02.3	**06.4**	02.3	**05.5**
+LogP +QED	l l	15.7	**32.1**	15.7	**26.7**	15.7	**24.4**	31.0	**61.9**	31.0	**56.7**	31.0	**57.5**
	s s	07.3	**12.7**	07.3	**14.8**	07.3	**14.9**	14.0	**38.8**	14.0	**37.7**	14.0	**38.2**
+LogP -QED	l l	14.4	**22.0**	14.4	**24.4**	14.4	**21.6**	20.3	**21.5**	20.3	**22.5**	20.3	**23.3**
	s s	03.9	**10.0**	03.9	**08.6**	03.9	**08.7**	03.9	**06.8**	**03.9**	03.8	03.9	**07.9**
-LogP +QED	l l	11.8	**28.8**	11.8	**23.8**	11.8	**29.2**	33.0	**59.0**	33.0	**56.5**	33.0	**54.9**
	s s	05.6	**10.1**	05.6	**08.0**	05.6	**08.9**	17.9	**34.2**	17.9	**35.7**	17.9	**32.7**
-LogP -QED	l l	07.8	**12.2**	07.8	**11.3**	07.8	**11.6**	19.7	**28.4**	19.7	**23.3**	19.7	**22.5**
	s s	**03.8**	02.8	**03.8**	02.8	**03.8**	02.5	05.6	**05.8**	05.6	**05.9**	**05.6**	05.3
Average	l l	11.8	**20.5**	11.8	**19.6**	11.8	**19.9**	24.3	**40.7**	24.3	**38.3**	24.3	**39.0**
	s s	06.5	**11.3**	06.5	**11.5**	06.5	**11.7**	12.8	**26.0**	12.8	**25.5**	12.8	**26.4**

Table 7: The results of AgentDrug with various sizes of the database D.

p	*d*	Qwen2.5-3B			Qwen2.5-7B		
		5K	10K	20K	5K	10K	20K
		accuracy	accuracy	accuracy	accuracy	accuracy	accuracy
+LogP	l	47.7	47.3	55.9	67.2	66.7	68.1
-LogP	l	36.6	38.8	41.0	77.3	71.5	73.0
+TPSA	l	41.8	41.2	43.6	64.5	62.8	65.0
-TPSA	l	46.2	45.7	52.0	69.8	65.3	64.5
Average	l	43.1	43.2	48.1	69.7	66.6	67.6
+LogP +TPSA	l l	28.0	31.7	21.2	49.6	46.2	51.6
+LogP -TPSA	l l	20.1	19.7	22.1	23.4	25.6	24.8
-LogP +TPSA	l l	18.9	29.6	22.6	65.0	54.8	55.2
-LogP -TPSA	l l	09.3	07.0	05.8	14.1	10.8	08.9
Average	l l	19.1	22.0	17.9	38.0	34.4	35.1

D Related Work

Recent advancements in AI, particularly in large language models (LLMs), have begun to transform the drug discovery process by efficiently scaling many core molecular tasks (Mak and Pichika, 2019). Notably, Seidl et al. (2023); Zhao et al. (2023); Liu et al. (2024b) have incrementally pretrained LLMs to advance molecular property prediction, synthesis, and retrosynthesis. In addition, Edwards et al. (2022); Liu et al. (2023b); Li et al. (2024); Le et al. (2024) have trained LLMs for molecular captioning. However, molecular optimization, which plays a critical role in designing improved compounds, remains underexplored.

Traditionally, supervised learning has been the dominant approach for addressing this task. For example, He et al. (2021, 2022) construct datasets containing millions of labeled molecule pairs and train models from scratch. Building on this foundation, Wu et al. (2024) incorporate external atomic embeddings into model training, while Turutov and Radinsky (2024) introduce a patentability penalty to guide the learning process. Additionally, Nahal et al. (2024) propose a framework that involves chemists directly in the training loop to improve model alignment with expert knowledge. In contrast to these supervised approaches, recent work explores the use of large language models (LLMs) to bypass the need for labeled molecule pairs (Liu et al., 2023a). LLMs offer strong zero-shot and open-vocabulary generalization capabilities, allowing them to operate beyond fixed objective sets. As a result, they present a promising alternative for molecular optimization while substantially reducing the cost and complexity of training.

As early pioneers, Zhang et al. (2024) investigate the use of LLMs for molecular optimization through straightforward prompting, but observe limited accuracy. Inspired by the refinement strategies commonly employed by chemists, Liu et al. (2024a) propose ChatDrug, a simple chat-based framework that guides LLMs through a refinement loop. In this loop, the LLM receives generic feedback such as "The modified molecule does not meet the objective" to revise the output toward the desired properties. However, relying solely on generic feedback leads the LLM to refine molecules in a largely unguided manner. Moreover, ChatDrug does not address the issue of molecular hallucination (Guo et al., 2023), where the LLM generates chemically invalid structures. As a result, the retrieval step is frequently disabled due to the high frequency of invalid outputs, limiting its overall effectiveness.

E Discussion

In this section, we give a discussion on the motivation and nature of molecular optimization. Despite the existence of huge chemical databases like ZINC, ChEMBL, PubChem, or commercial catalogs containing billions of molecules, design and optimization are still essential for several reasons:

- The estimated drug-like chemical space is $> 10^{60}$ molecules. Even huge databases like ZINC, ChEMBL, and PubChem are just a small fraction of that space. Thus, the chance that the target molecule (satisfying multiple objectives) is already enumerated is low. Moreover, retrieval over huge molecule databases is extremely expensive because computing similarity between molecules is hard to parallelize.
- Novelty for patentability. Pharmaceutical companies need novel chemical compounds they can patent. Retrieving a molecule already reported in the literature often means that they cannot claim exclusivity, making it commercially unviable.

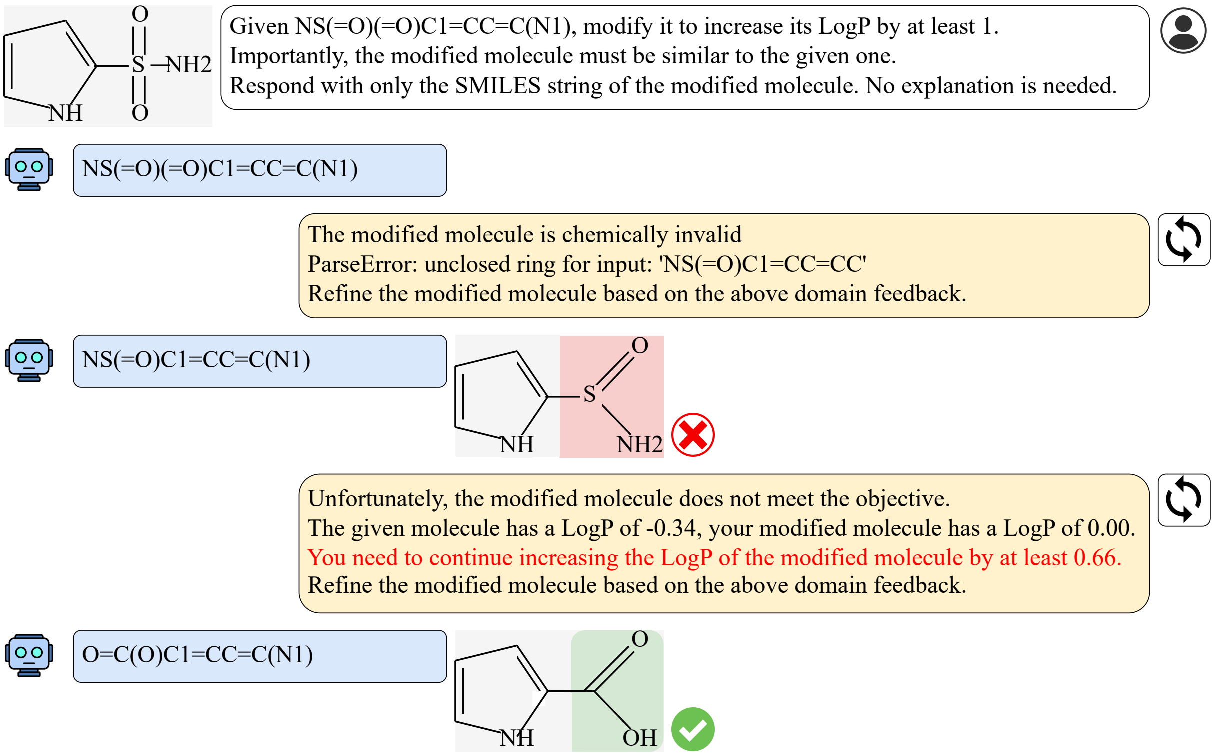
In summary, retrieval from databases is only a valuable starting point to augment design and optimization. For example, AgentDrug leverages a small database (10K molecules) to augment the optimization process. It depends on the size of the database; the molecule m_e can be easier or harder to find. Also, the retrieval may fail or return a low-similarity molecule in sparse areas.

F Full Algorithm

```
Algorithm 1: AgentDrug
Require: a given molecule $m$, an LLM $\mathcal{M}$.
$\widehat{m} = \mathcal{M}(p||d||m)$                                  ▷ initiated
For iteration in 1, ..., T
    is_valid, ParseError $= \mathcal{T}(\widehat{m}, \text{parsing})$
    If is_valid $= 0$ then
        $\widehat{m} = \mathcal{M}(p||d||m||\widehat{m}||\text{ParseError})$
    Else
        If $\prod_{i=1}^{N} \underset{p_i, d_i}{E}(\widehat{m}) = 1$ then
            break
        Else
            $\widehat{m} = \mathcal{M}(p||d||m||\widehat{m}||\nabla_p||m_e)$
Return $\widehat{m}$
```

G Examples

Illustrative examples of AgentDrug's workflow, including detailed prompts, are provided in Figure 5 and Figure 6. The *gradient* signal used to guide the LLM is highlighted in red for clarity.



Given NS(=O)(=O)C1=CC=C(N1), modify it to increase its LogP by at least 1.
Importantly, the modified molecule must be similar to the given one.
Respond with only the SMILES string of the modified molecule. No explanation is needed.

NS(=O)(=O)C1=CC=C(N1)

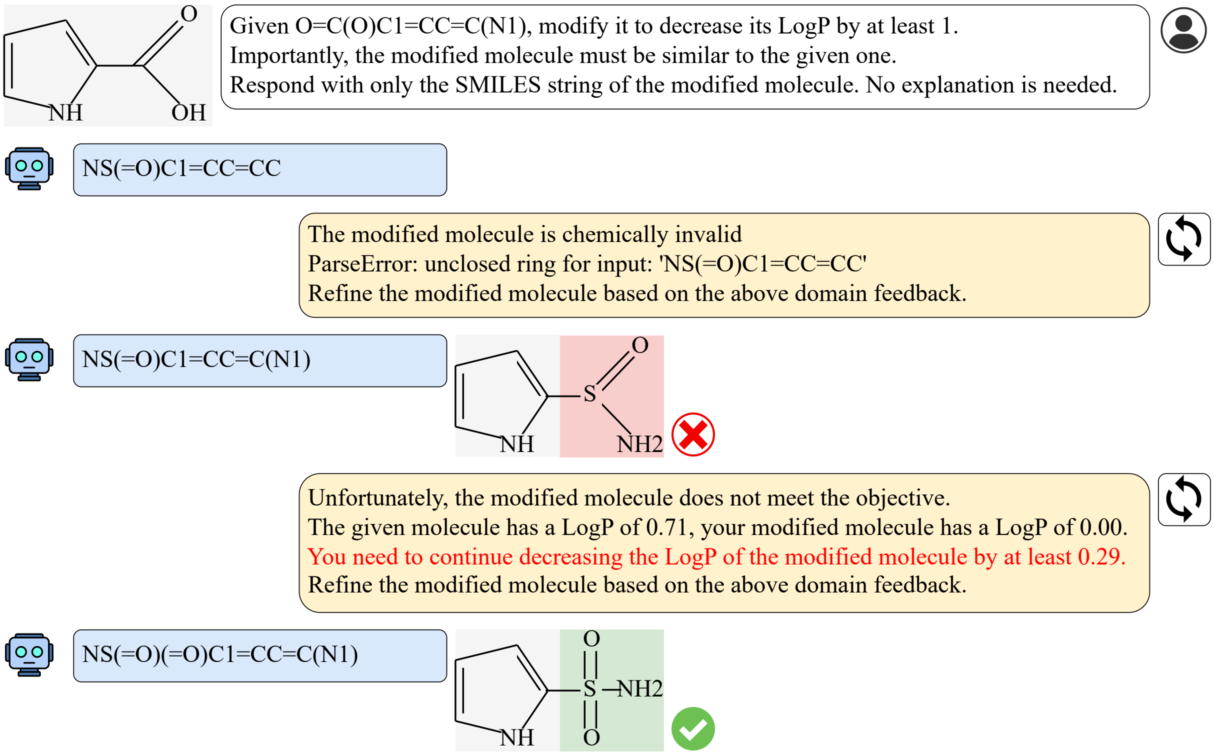
The modified molecule is chemically invalid
ParseError: unclosed ring for input: 'NS(=O)C1=CC=CC'
Refine the modified molecule based on the above domain feedback.

NS(=O)C1=CC=C(N1)

Unfortunately, the modified molecule does not meet the objective.
The given molecule has a LogP of -0.34, your modified molecule has a LogP of 0.00.
You need to continue increasing the LogP of the modified molecule by at least 0.66.
Refine the modified molecule based on the above domain feedback.

O=C(O)C1=CC=C(N1)

Figure 5: An example of how AgentDrug works, including detailed prompts.



Given O=C(O)C1=CC=C(N1), modify it to decrease its LogP by at least 1.
Importantly, the modified molecule must be similar to the given one.
Respond with only the SMILES string of the modified molecule. No explanation is needed.

NS(=O)C1=CC=CC

The modified molecule is chemically invalid
ParseError: unclosed ring for input: 'NS(=O)C1=CC=CC'
Refine the modified molecule based on the above domain feedback.

NS(=O)C1=CC=C(N1)

Unfortunately, the modified molecule does not meet the objective.
The given molecule has a LogP of 0.71, your modified molecule has a LogP of 0.00.
You need to continue decreasing the LogP of the modified molecule by at least 0.29.
Refine the modified molecule based on the above domain feedback.

NS(=O)(=O)C1=CC=C(N1)

Figure 6: An example of how AgentDrug works, including detailed prompts.